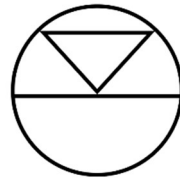


# Linn Crestron 4-Series Driver for SIMPL Windows

Installation Guide

Author: Cameron Smith  
[cameron.smith@linn.co.uk](mailto:cameron.smith@linn.co.uk)

June 2023



**LINN**



## Contents

Introduction .....	4
Requirements.....	5
Driver Overview .....	6
Demo Project .....	8
Control Point GUI Setup.....	8
Making Modifications to the Driver.....	8
Signal Suffix Types.....	9
<i>LinnDriver</i> Signal List .....	10
Room Section .....	10
Module Parameters .....	11
<i>LinnControlPoint</i> Signal List.....	12
Section A – CCP Information, Room/Source Information & Standby.....	12
Section B – Volume & Transport Control.....	13
Section C – Seeking & Metadata.....	14
Section D – Pins.....	15
Section E – Room & Source Selection .....	15
Module Parameters .....	16
Design Tips .....	17
Room Selection Page .....	17
Pin Usage.....	17
External Source Invisibility .....	18
Multi-Byte and International Fonts .....	18
Driver Enable and Refresh Buttons.....	18
Single Room Setups.....	18
Macro Commands.....	19
Add-on Modules .....	20
Airable Radio – <i>LinnAirable</i> Module .....	21
Browse Page.....	22
Search Page.....	24
Actions Page.....	26
TIDAL – <i>LinnTidal</i> Module .....	28
Home Page.....	29
Browse Page.....	31
Search Page.....	34
Collection Page.....	37

Actions Page.....	39
Qobuz – <i>LinnQobuz</i> Module.....	41
Home Page.....	42
Other Pages.....	43
Playlist – <i>LinnPlaylist</i> Module.....	44
Playlist .....	45
Edit .....	47
Glossary.....	49

## Introduction

This document aims to provide guidance in setting up a Crestron project that utilises the *Linn Crestron 4-Series Driver for SIMPL Windows*. This set of SIMPL modules offers an interface for network control over several Linn DS/DSM devices including the Linn Kustom DSM. The driver operates by utilizing Linn's custom-install **Gateway API\*** software. Rather than handling one-to-one communication between Crestron processor and (potentially several) Linn devices the gateway API arranges all Linn products on a local network into a **house topology** and manages control. By default, the maximum number of **rooms** allowed by this driver in a setup is 20. This gateway API is available on any **Core4** Linn device (see <https://docs.linn.co.uk/wiki/index.php/Category:CORE4>) or by having a device running Kazoo Server on the same network. Only one instance of the gateway is required. The Gateway API is enabled on Linn devices through the advanced settings in Linn's 'manage systems' web configuration. The Gateway API will be automatically discovered on the local network on start-up. Control over the devices is then accessed through a **Crestron control point (CCP)**. A list of available features is given below:

- Room selection - each CCP has an actively selected room within the house topology that can be changed via the CCP itself.
- Grouping - the current **grouping** of rooms set via Kazoo/LinnApp will be reflected in the room selection menu and automatically update whenever a change in grouping occurs.
- For each room/device in the house, the user has control over the following DS functionalities:
  - Standby control – each room can be individually toggled between standby states.
  - Volume control – step up/down, set to absolute value, and toggle mute.
  - Toggle shuffle and repeat states for appropriate sources.
  - Transport control over any active playlist – play, pause, play/pause toggle, skip previous, skip next.
  - Source selection – capable of selecting external sources on a DS.
  - Metadata – view metadata for the currently playing item including artwork.
  - Seeking – view the time elapsed, time remaining, and the time progress on a gauge bar for the active track. Input to move to a new time position within the active track.
  - Pin selection – as on physical Linn DSM devices, customizable buttons can be set up using a Linn control point (Kazoo/LinnApp) to provide easy access to favourite radio stations, playlists, or external sources. The name and associated artwork for each pin can also be accessed via the module and displayed on a CCP interface.
- Standby states of all rooms within the house can be changed quickly and directly from a room selection menu without having to select the room as the **active room** of a CCP first
- Through SIMPL Windows, a CCP can be set to lock onto a certain room if that room is found through the gateway API. Useful for systems where perhaps only 1 Linn DS is in use and allows room selection to be omitted from the CCP.
- For more advanced macro functionality, a string input optionally allows programmers to run **LPEC** commands directly to devices in the house from the Crestron processor.

This driver is built in a modular format to allow for additional functionality, including browsing/ searching of third-party streaming services. These are added in separate SIMPL modules which can be added/omitted from the setup as per the customer's requirements. The currently available *add-on* modules:

- Airable radio – browsing of airable radio stations with a search feature. Once found stations can be selected for playback, favourited/unfavourited, or assigned to pins on the DS of the active room.
- TIDAL – browsing of the TIDAL streaming service catalogue with a search feature. Tracks, albums, and playlists can be selected for playback, favourited/unfavourited, or assigned to pins on the DS of the active room.
- Qobuz – browsing of the Qobuz streaming service catalogue with a search feature. Tracks, albums, and playlists can be selected for playback, favourited/unfavourited, or assigned to pins on the DS of the active room.
- Playlist – each DS has its own playlist (or queue) which tracks can be added to from various sources. The playlist module allows for this to be viewed for the active room and any track can be selected for playback. The playlist can also be cleared or modified by moving tracks around or removing them individually.

This list of additional modules will be expanded in the future with Songcast groupings and Linn cloud playlists being the next two scheduled for development.

Note: this driver is designed for use in SIMPL Windows and is not a *Crestron Driver* designed for use with the Crestron Home OS.

## Requirements

Access to Crestron's pool of software (SIMPL Windows, Toolbox, Vision Tool Pro-e, etc) is required for setting up a system. **A Crestron 4-series control processor is also required.** Earlier Crestron control processors (3-series and before) are not compatible for use with this driver.

An intermediate level of Crestron installation experience is required for the setup and integration of this module into a project; particularly in working with SIMPL Windows and Vision Tools Pro-e (or HTML5). No SIMPL+ or SIMPL# programming experience is required for a basic installation.

An understanding of Linn DS/DSM product functionality is also assumed including the Kustom DSM with its stream and zone architecture. The Kustom DSM is a product by Linn designed for use within custom installs and will therefore commonly be used in conjunction with this driver (for further information see: [https://docs.linn.co.uk/wiki/index.php/Kustom\\_DSM](https://docs.linn.co.uk/wiki/index.php/Kustom_DSM))

## Driver Overview

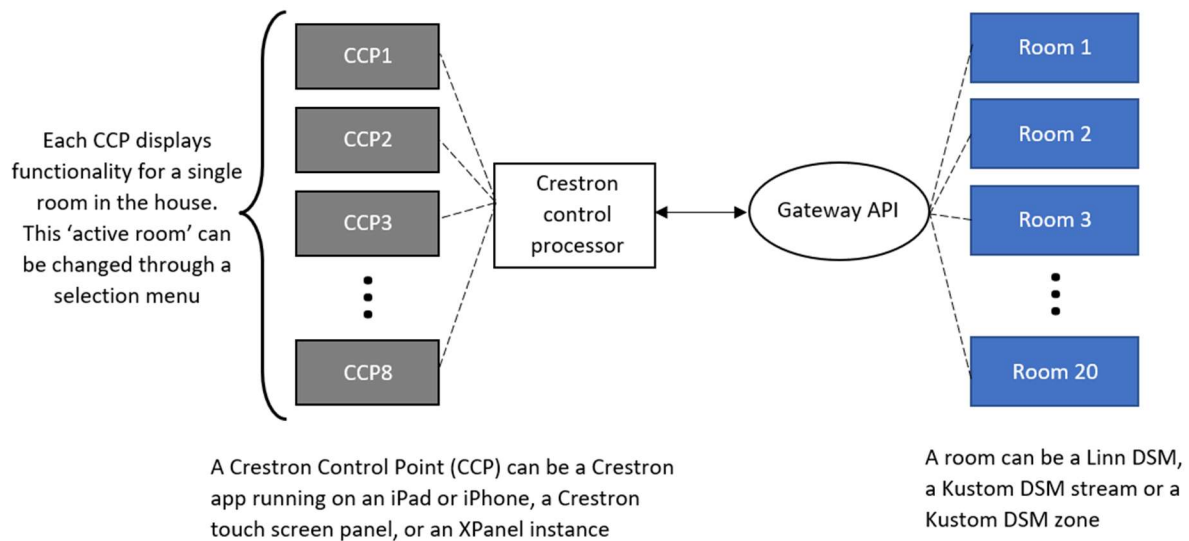


Fig. 1 – Overview of a setup using the Crestron Linn Driver for SIMPL Windows. By default, there can be up to 20 Linn 'rooms' on the local network that the Gateway API will organise and communicate with. The Crestron control processor will communicate and receive any changes that occur with these devices. You can add as many Crestron control points (CCP) as needed for the devices you have available (Touch panels, iPads etc). These will be located around the house and are used to control the functionality of these rooms. Each CCP will have one active room selected at any time that can be changed through the room selection menu.

An overview of the basic structure of a setup incorporating this driver can be seen in Fig. 1. The driver itself has two core SIMPL modules/symbols that need to be incorporated in a SIMPL Windows project:

- **LinnDriver**

- This is the 'brain' of the driver that is responsible for establishing communication with the Gateway API and managing information received from it to update CCPs accordingly.
- The SIMPL symbol's interface is shown in Fig. 2. The module is headed by essential inputs and outputs. The macro command input that allows for more advanced scripting of macros also appears here.
- The following section, minimized in Fig. 2, contains information shared across all CCPs about the rooms within the house. This information is used by the room selection menu which contains the names of every room and individual, toggleable standby control buttons.
- The module/symbol contains several parameters. Some of these are optional such as overriding the discovery mechanism by entering the gateway API IP/port manually. The processor IP and image paths are required for the image files of source types. All these are explained in greater detail in the section 'Module Parameters'.

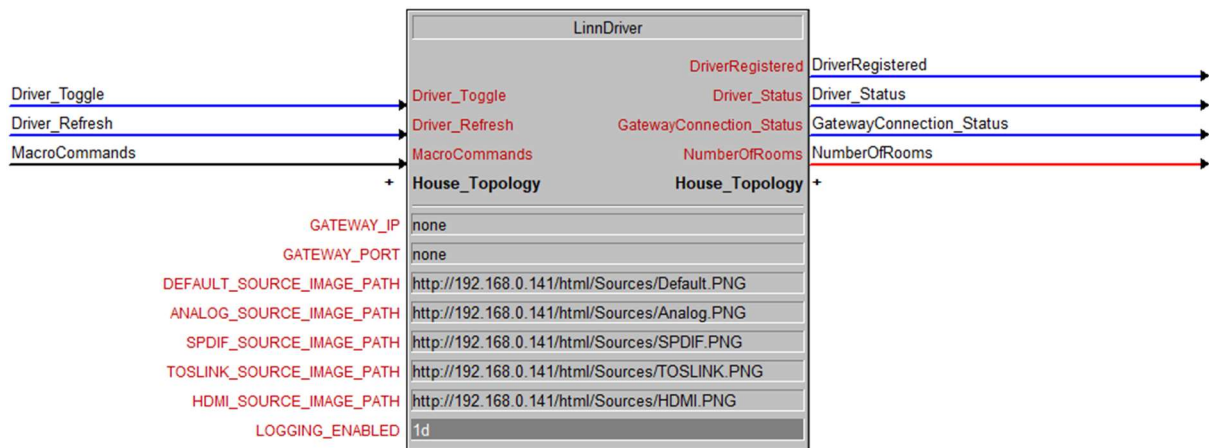


Fig. 2 – Overview of the LinnDriver SIMPL module. Some basic information about the program and gateway connection state as well as the macro command input and the number of rooms in the active topology are given at the top. This is followed by a section containing shared information about each room in the house. At the bottom, a number of parameter fields are also visible which will be explained in detail.

- **LinnControlPoint**

- This SIMPL modules provides all the unique input and output signals required within a CCPs GUI. One of these must be added for each CCP used in the house setup, as shown in Fig. 3, and given a unique name in the module parameter. Each CCP is able to select an active room from the list of devices on the local network.
- On start-up of the program these modules will automatically register themselves with the main driver module. No connection of signals between the *LinnDriver* module and the *LinnControlPoint* module is required.
- The list of input and output signals contained in the module covers all the basic functionality outlined in the introduction. A description of all these input and output signals is given in Section ‘*LinnControlPoint* Signal List’.

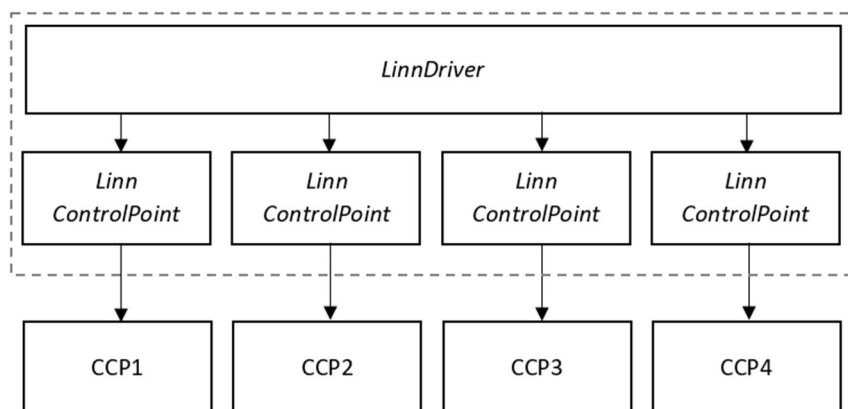


Fig. 3 – Structure and connection between the two core modules that make up the program. Control Point modules only need to be added if they are necessary in the project setup: for example, the above setup only contains 4 CCPs and therefore only uses 4 instances of the *LinnControlPoint* module.

## Demo Project

A demo project is provided alongside this driver's basic components. The example GUI design .vtp file is not intended to be used in a Crestron customer setup but rather to be a point of reference and for helping understand how to implement this driver in a project. It is expected that other home automation aspects unique to the customer will also be included in the project and controlling Linn products will not be the sole function of the Crestron system (although it is certainly possible). This driver is designed to work alongside these other features.

**One particular facet of the demo project that it is highly recommended to follow is the use of digital, analog, and string joins from the VisionTools Pro-e project.** A non-exhaustive list of examples where following the demo project signal joins becomes useful:

1. Transport controls will become greyed out when not applicable to the active source.
2. Standby buttons on the room selection menu will be hidden where they are irrelevant.
3. Loading processes will be made visually cleaner.

The demo contains a setup that includes 2 CCPs under ethernet devices: the first being an XPanel instance and the second being an iPad. These both utilise identical GUI projects (*Linn\_Demo\_XPanel.vtp* and *Linn\_Demo\_iPad.vtp* for respective device types) however this does not have to be the case in a customer project.

To import the project, open SIMPL Windows then *File->Import Archived Program...* and select the .zip file. The program can then be compiled and loaded to your 4-series Crestron processor. The VisionTools Pro-E projects will also have to be opened, modified with the IP address of the Crestron processor in use, and built.

## Control Point GUI Setup

A GUI will have to be designed for use with this driver. This is completely customisable and should be tailored to the Crestron customers unique requirements but will generally feature a similar structure. The functionality in the demo project is split into some basic pages:

- Home page
- Now playing page
- Room selection page
- Source selection page
- Pin selection page
- Pages for any added services (explained further in dedication sections)
- Settings page

The home page contains *page flip* references to the other pages and can be returned to at any point by pressing the home button. In a customer's project this is fully flexible and be changed to the designers liking. Some tips towards building a GUI project that includes this driver are detailed later in this document.

## Making Modifications to the Driver

The SIMPL# code is precompiled, and it is not possible to make changes. Uninformed changes to the SIMPL+ source code files will likely cause serious problems in the user experience. However, some aspects such as the maximum room count can be modified with some Crestron programming



experience. To increase the maximum number of rooms, open the SIMPL+ files for *LinnDriver.usp* and *LinnControlPoint.usp* and change the value for the #DEFINE\_CONSTANT MAX\_ROOMS. After recompiling the SIMPL+ source files, the accompanying SIMPL Windows modules will have to be adjusted to accommodate the extra signals. For further information, advice or to request modifications, please contact: [helpline@linn.co.uk](mailto:helpline@linn.co.uk).

### Signal Suffix Types

Some general conventions for suffixes are used in the naming of module inputs and outputs and are outlined below.

Suffix	Type	Description
_Toggle	Digital Input	An input which will toggle a functionality.
_Select	Analog Input	An input to the module used to select an item in a list.
_Status	Output (Any type)	An output which gives feedback of the current state of a GUI feature. For example, the digital high or low state of a mute button, or a string containing track metadata.
_Available	Digital Output	Output signals used to make buttons controlling corresponding inputs unavailable. For example, depending on the current source of a DS features like shuffle and repeat may not be relevant and hence should not be available for selection in the CCP.
_Visible	Digital Output	Output signals used to make buttons controlling corresponding inputs invisible in certain situations.

## LinnDriver Signal List

Beginning at the top of the LinnDriver symbol, previously shown in Fig. 2, there are signals for the general operation of the driver.

### Inputs

Name	Type	Description
Driver_Toggle	Digital	Toggles the driver between being in a disabled or active state. Defaults to the enabled state on loading to the Crestron processor.
Driver_Refresh	Digital	If program is active, this will refresh the program. All known data about Linn devices and their states will be reset and the current Gateway API session will be closed and re-established.
MacroCommands	String	Used for sending LPEC commands directly to rooms within the house. Optional advanced feature, explained in greater detail in <i>Macro Commands</i> section.

### Outputs

Name	Type	Description
DriverRegistered	Digital	On loading the program, the driver will register itself. If not high, then an error has occurred.
ProgramActive_Status	Digital	Indicates the current active state of the driver. If active this will be high in value, else it will be low.
GatewayConnection_Status	Digital	Indicates the current state of the connection with the gateway API. If a connection is open this will be high in value, else it will be low.
NumberOfRooms	Analog	Once a connection to the gateway API is established this signal will output the number of rooms (more specifically, the number of lines required in a list of rooms within the house).

## Room Section

The following sets of signals provide information about the rooms within the house. This information is shared across all CCPs and will be used in the room selection menu of every CCP.

### Inputs

Name	Type	Description
Room[1-20]_Standby_Toggle	Digital	Directly toggle the standby state of the corresponding room.

### Outputs

Name	Type	Description
Room[1-20]_Standby_Status	Digital	Give the standby state of the rooms in the current house setup. Note: Digital signal high for standby off (i.e., room turned on), low for standby on.
Room[1-20]_Standby_Visible	Digital	Defines the visibility of the room standby buttons. In some cases, with the Kustom DSM, a standby button should not be visible.
Room[1-20]_Text	String	The names of the rooms in the current house setup.

## Module Parameters

Name	Description
GATEWAY_IP	Optional: By entering a valid IPv4 address into this field, the automatic discovery mechanism will be overridden, and this address will be used instead.
GATEWAY_PORT	Optional: By entering a valid port number (integer 1025-65536) into this field, the default gateway port number of 4100 will be overridden.
DEFAULT_SOURCE_IMAGE_PATH	Enter the local path for the image that will be used for source that don't conform to any of the following types.
ANALOG_SOURCE_IMAGE_PATH	Enter the local path for the image that will be used for external sources of the analog type.
SPDIF_SOURCE_IMAGE_PATH	Enter the local path for the image that will be used for external sources of the SPDIF type.
TOSLINK_SOURCE_IMAGE_PATH	Enter the local path for the image that will be used for external sources of the TOSLINK type.
HDMI_SOURCE_IMAGE_PATH	Enter the local path for the image that will be used for external sources of the HDMI type.
LOGGING_ENABLED	Use to enable logging on the Crestron console for debugging purposes. <i>0d</i> for disabled <i>1d</i> for enabled.

The set of parameters for source image paths are used to provide appropriate images on-screen for the various external source categories. For the external sources of a Linn DSM there is no standard metadata artwork image that is used. This is left up to the installer to choose with the aim that images are chosen that fit with the theme of the Crestron project GUI.

The first step is to take the image files and host them locally on the Crestron processor using FTP or Crestron Toolbox. Once transferred the paths for these image files can be obtained. These will take the form of "<http://192.168.0.141/html/Sources/Analog.PNG>" depending on their location. These paths can now be inserted in the parameter fields.

## LinnControlPoint Signal List

This module acts to modularise the functionality that is common across CCPs, keeping the *LinnDriver* module's interface clean. All inputs from buttons and feedback to the GUI for basic functionality is made available through this SIMPL module. One of these should be added for each CCP that is used in the project configuration. The signal list will be detailed in full here.

### Section A – CCP Information, Room/Source Information & Standby

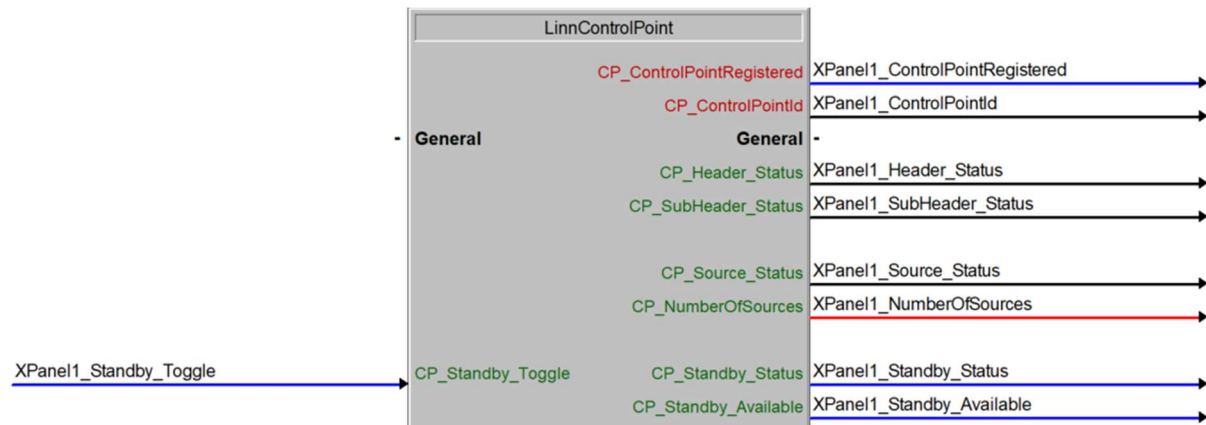


Fig. 3 – Screenshot of the first portion of signals in the *LinnControlPoint* SIMPL module

#### Inputs

Name	Type	Description
CP_Standby_Toggle	Digital	Toggle the standby state of the active room.

#### Outputs

Name	Type	Description
CP_ControlPointRegistered	Digital	On loading the program, the control point will register itself. If not high, then an error has occurred.
CP_ControlPointId	String	The name of the control point as set through the CONTROL_POINT_NAME parameter.
CP_Header_Status	String	Primary information about the current state of the CCP. It will display text when searching for the gateway connection or loading and give information about which room within the house topology is currently selected by the CCP.
CP_SubHeader_Status	String	Secondary information about the current state of the CCP. Used when a <b>listening room</b> is selected, the listening room's name will be output through this signal.
CP_Source_Status	String	The current source selected by the active room.
CP_NumberOfSources	Analog	The number of sources for the active room. Used by the source selection menu subpage to determine the number of items in the list.
CP_Standby_Status	Digital	Feedback for the standby state of the active room. Note: Digital signal high for standby off (i.e., room turned on), low for standby on.
CP_Standby_Available	Digital	Availability of the standby control button. For cases where no room is actively selected the standby button in the CCP should become unavailable.

## Section B – Volume & Transport Control

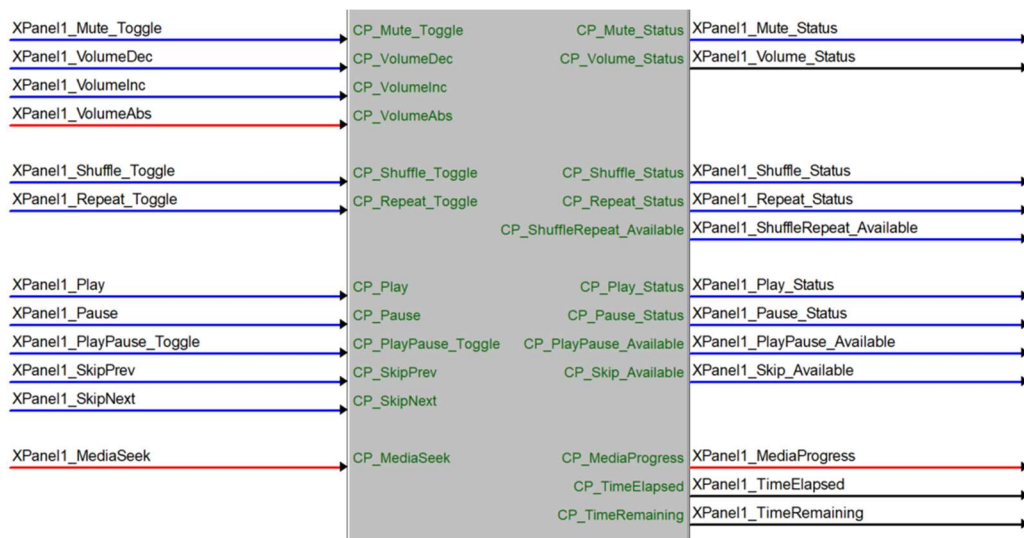


Fig. 4 – Screenshot of the second portion of signals in the *LinnControlPoint* symbol

### Inputs

Name	Type	Description
CP_Mute_Toggle	Digital	Toggle the mute state of the active room.
CP_VolumeDec	Digital	Decrease the volume of the active room by 1. Held presses will cause the volume to ramp at a rate of 4/second.
CP_VolumeInc	Digital	Increase the volume of the active room by 1. Held presses will cause the volume to ramp at a rate of 4/second.
CP_VolumeAbs	Analog	Sets the volume of the active room to an absolute value. Any time a change is made to this signal the room's volume will be set to that value.
CP_Shuffle_Toggle	Digital	Toggle the shuffle state of the active room.
CP_Repeat_Toggle	Digital	Toggle the repeat state of the active room.
CP_Play	Digital	Set the transport state of the active room to play.
CP_Pause	Digital	Set the transport state of the active room to pause.
CP_PlayPause_Toggle	Digital	Toggle the transport state of the active room between play and pause. Alternative to individual play and pause buttons.
CP_SkipPrev	Digital	Skip backward to the previous item in the playlist of the active room.
CP_SkipNext	Digital	Skip forward to the next item in the playlist of the active room.

### Outputs

Name	Type	Description
CP_Mute_Status	Digital	Feedback for the mute state of the active room.
CP_Volume_Status	String	The current volume level of the active room.
CP_Shuffle_Status	Digital	Feedback for the shuffle state of the active room.
CP_Repeat_Status	Digital	Feedback for the repeat state of the active room.

CP_ShuffleRepeat_Available	Digital	Availability of the shuffle and repeat toggle buttons. For cases where these buttons are irrelevant for the actively selected room's source these will become unavailable in the CCP.
CP_Play_Status	Digital	Feedback for the play state of the active room.
CP_Pause_Status	Digital	Feedback for the pause state of the active room.
CP_PlayPause_Available	Digital	Availability of the play and pause buttons. For cases where these buttons are irrelevant for the actively selected room's source these will become unavailable in the CCP.
CP_Skip_Available	Digital	Availability of the skip prev and skip next buttons. For cases where these buttons are irrelevant for the actively selected room's source these will become unavailable in the CCP.

## Section C – Seeking, Metadata & Error

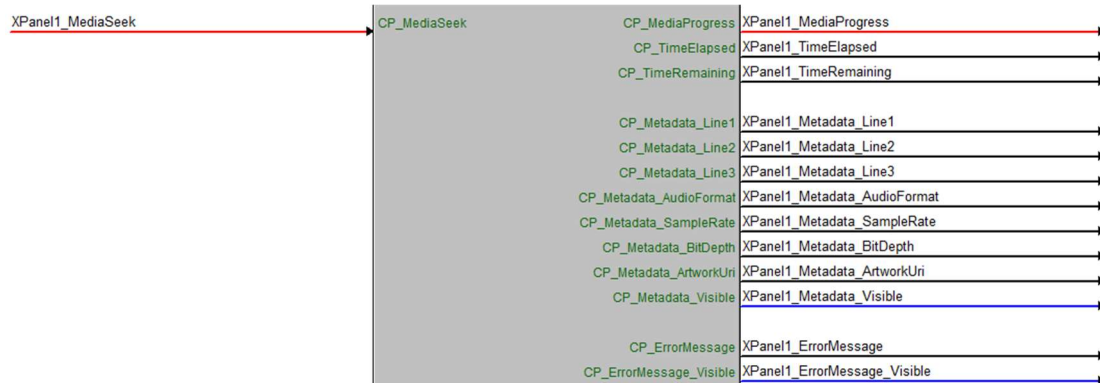


Fig. 5 – Screenshot of the third portion of signals in the *LinnControlPoint* SIMPL module

### Inputs

Name	Type	Description
CP_MediaSeek	Analog	For relevant sources, move to a new position in the currently playing track. Use an integer value in the range 0-255, this will be mapped as a percentage.

### Outputs

Name	Type	Description
CP_MediaProgress	Analog	Feedback for the progress of the currently playing track. Progress is mapped as an integer value between 0-255.
CP_TimeElapsed	String	The time elapsed for the currently playing item (uses format hh:mm:ss).
CP_TimeRemaining	String	The time remaining for the currently playing item (uses format hh:mm:ss).
CP_Metadata_Line1	String	The first line of metadata for the active room. Commonly, a track title, radio station name, or external source name.
CP_Metadata_Line2	String	The second line of metadata for the active room. Commonly, an artist name or radio metatext.
CP_Metadata_Line3	String	The third line of metadata for the active room. Commonly an album name.

CP_Metadata_AudioFormat	String	The audio format for the active room.
CP_Metadata_SampleRate	String	The sample rate for the active room.
CP_Metadata_BitDepth	String	The bit depth for the active room.
CP_Metadata_ArtworkUri	String	The artwork URI for the active room.
CP_Metadata_Visible	Digital	Visibility of the metadata. Hides metadata during standby and loading processes.
CP_ErrorMessage	String	Info about the error that has occurred.
CP_ErrorMessage_Visible	Digital	Used to hide the error message info when no error has occurred.

## Section D – Pins

### Inputs

Name	Type	Description
CP_Pin[1-6]	Digital	Select the corresponding pin for playback for the active room.

### Outputs

Name	Type	Description
CP_Pin[1-6]_Available	Digital	Availability of the corresponding pin buttons. For cases when a pin is empty, loading or no room is currently selected, the pin will be unavailable.
CP_Pin[1-6]_Name	String	The name of the item the pin contains for the active room.
CP_Pin[1-6]_ArtworkUri	String	The artwork URI of the item the pin contains for the active room.

## Section E – Room & Source Selection

These final sets of signals are related to the room and source selection pages.

### Inputs

Name	Type	Description
CP_Room[1-20]_Select	Digital	Select a room from the room list to become the new active room.
CP_Source[1-20]_Select	Digital	Select a source from the source list of the active room to become active.

### Outputs

Name	Type	Description
CP_AnyRoom_Selected	Digital	Indicate if any room is actively selected on the CCP. Digital high if so, low if not.
CP_Room[1-20]_Selected	Digital	Whichever room in the room list is currently active for the CCP will be shown in this set of digital outputs. Gives feedback for the room selection list by allowing it to highlight the active room.
CP_Source[1-20]_Selected	Digital	Whichever source is currently selected for the active room will be shown in this set of digital outputs. Gives feedback for the source selection list by allowing it to highlight the active source.
CP_Source[1-20]_Text	String	The names of the available sources for the active room. Used by the source selection list.

## Module Parameters

Name	Description
CONTROL_POINT_NAME	A name that describes the control point (eg. Hallway TP, Living Room iPad). Must be unique from other control point module instances. Used for debugging purposes and for associating additional service modules.
DEVICE_LOCK	Optional: By entering a name (or <b>UDN</b> ) of a device/room the control point will automatically select that room if it is present in the house. <b>If no device lock is in use enter "none" as the parameter value.</b>



## Design Tips

### Room Selection Page

The room selection page from the demo project is shown in Fig. 6. This page was built in VisionTools Pro-e and uses a 'Crestron Subpage Reference List' item. This object takes in a reference to another subpage (a list item) and forms a list of these. This is to allow for standby buttons for each room in the house topology to be present next to the room names. Care must be taken when designing a similarly presented room selection feature for a customer's project. Pay specific attention to the way that digital join increments are used and how the formatted text labels contain the string joins.

It can also be seen in Fig. 6 that rooms are presented in a list with any existing groupings shown via use of indentation. Rooms within the house are ordered automatically and these indentations are prepended to the output strings. Also note the invisibility of standby buttons for Kustom streams within a group led by itself. This is simply to avoid multiple buttons which serve identical functionality.

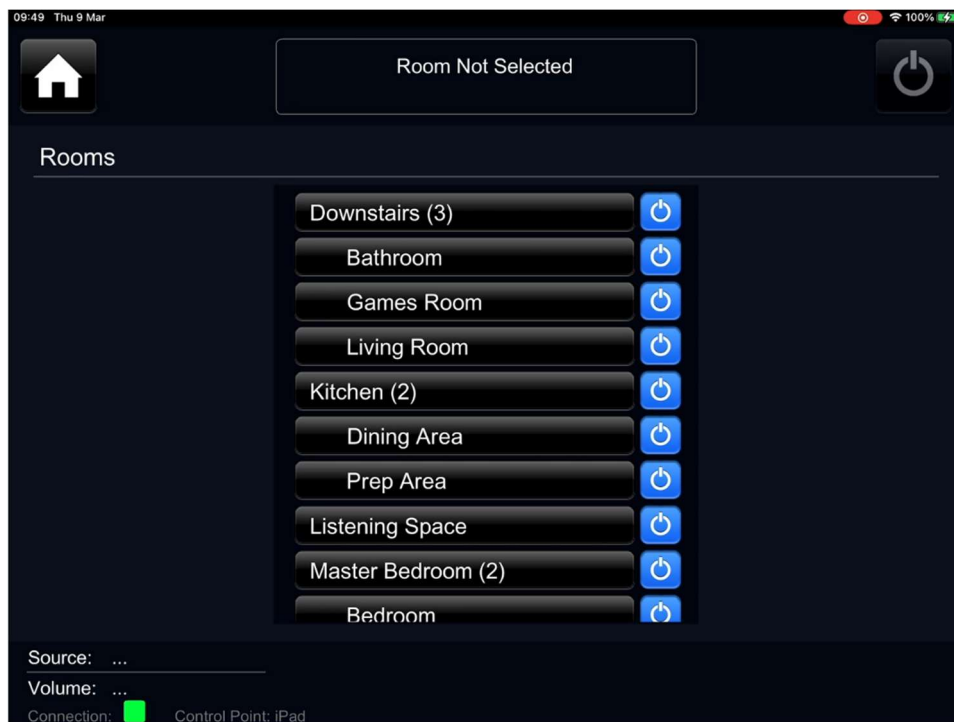


Fig. 6 – Screenshot of the room selection menu/sub-page for the demo project with an example house setup. The first group is a Kustom DSM stream with 3 listening zones. Further down a standalone DSM named *Listening Space* can be seen.

### Pin Usage

Pins can play an important role in a Crestron project utilising this driver. The digital inputs exposed through the module work in the same manner as the physical pins on Linn Majik/Selekt/Klimax DSM devices and causes them to load a media item set through a Linn control point (Linn Kazoo, Linn App) or through the driver itself.

These can be set to be external sources, radio stations, playlists, or albums. This allows the customer to quickly select from a set of favourite media items directly through a CCP. These pins should be configured during the setup process however a customer is free to reconfigure these themselves if they wish to and they will be automatically updated for the Crestron system.

## External Source Invisibility

Linn DS/DSM devices have a range of external inputs available however not all may actually be in use. It is recommended that unused ports are hidden via the configuration settings in the Manage Systems of the customer's Linn account. Ports can also be given more relevant names as to what they contain (i.e., 'HDMI1' could be renamed to 'PS5'). These settings will be reflected in the source selection menu of the CCP and keep the interface tidy.

## Multi-Byte and International Fonts

Working with metadata for music from streaming services will mean the occasional encountering of text which contains characters outside the ASCII character set (for example Chinese, Russian or Korean characters). There is support built into the driver to allow for these to be displayed. For this to work the theme "Standard Theme International.xml" must be used in a VisionTools Pro-e project. The font "Arial Unicode MS" must also be used for the text labels where this is relevant. This is demonstrated in the demo project included with this driver. For more information see the official Crestron documentation: [https://support.crestron.com/app/answers/detail/a\\_id/5264/kw/utf-8](https://support.crestron.com/app/answers/detail/a_id/5264/kw/utf-8).

## Driver Enable and Refresh Buttons

The *Driver\_Toggle* and *Driver\_Refresh* digital inputs do not need to be placed in obvious locations of the project interface as the driver is automatically enabled on start-up and will automatically attempt to reconnect when required. It is recommended that they are attached to a button somewhere on the CCP though. If your project contains a generic settings page, it could prove useful to include these buttons as an easy method for resetting the driver in the case of unexpected problems.

## Single Room Setups

For Crestron projects that implement this driver where there is only one Linn device in the setup there are additional advisory steps. For these setups there is no reason to utilise room selection and it is recommended that the device lock parameter for all CCPs is set to the single room's name (or UDN). The result of this is that the device will be automatically selected on start-up.

## Error Message

If an error occurs the details of the error will be shown in the *CP\_ErrorMessage* output signal. The associated visibility signal will also go momentarily high. In the demo project an error subpage is added on top of every page to display this error. This visibility output signal is used to hide it at all other times.

## Macro Commands

The string input *MacroCommands*, is included as a simple way to allow for more direct control of DS functionality. This input utilises the LPEC protocol that can be used to control Linn DSMs over a Telnet connection. Upon loading the driver, telnet sessions are automatically opened and setup for all available rooms and can then be used at any point to send LPEC commands. The format of messages sent to this string input should be of the form:

*<Room Name> LPEC Command*

For example, to set the volume of a DS:

*<Living Room> Action Ds/Volume 2 SetVolume "45"*

The driver will search for room that matches the name in the chevrons (angle brackets). If the room is found the LPEC command will be forwarded to that device, else it will be discarded.

The protocol does included scope for eventing (ie. service's events can be subscribed to and subsequent changes in the evented variables will cause unsolicited event messages to be sent from the device). This should not be used however as there is no handling of responses over the socket connection. Any changes caused by actions will cause a response from the gateway API that will correctly update the values within the setup. **Note: the LPEC protocol is not compatible for use with Kustom zones.** For further information about this protocol see:

<https://docs.linn.co.uk/wiki/index.php/Developer:LPEC>

A SIMPL+ module could be written to allow for macro a series of commands to be sent out from a single button press. A single button press could result in a series of commands being sent out to multiple devices. If doing so it is recommended that a small DELAY is placed between each setting of the string input. For example:

```
soMacroCommands = "<Living Room> Action Ds/Volume 2 SetVolume \"45\"";
DELAY(5);        // a 50ms delay
soMacroCommands = "<Dining Room> Action Ds/Volume 2 SetVolume \"65\"";
```

## Add-on Modules

With this new Crestron driver comes new modular support for more advanced functionality including the browsing of 3<sup>rd</sup> party services. There are currently modules available for the browsing and searching of the services Airable (radio), TIDAL, and Qobuz (music-streaming). There is also a module for viewing and editing the playlist (queue) of any Linn DS on the network. This support is optionally added to a project through a modular technique. In Fig. 7 a program view containing the logic of the demo project is shown. The services are added to each control point as an optional “add-on” SIMPL modules containing all the necessary signals.

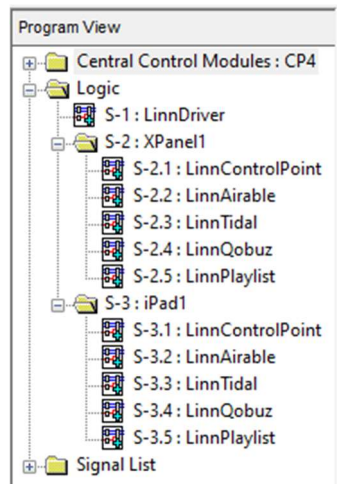


Fig. 7 – Screenshot of the program view for the demo project with a setup of two control points named “XPanel1” and “iPad1”. Alongside each control point module there are also add-on modules for Airable, TIDAL, Qobuz, and the DS playlist which contain all the signals required for these services/features.

This modular technique means that only the services/features required by the customer need to be added to the project and can be added/omitted on a per room basis. For example, you might only require full browsing of a streaming service for one room in the project. To associate a *LinnControlPoint* module with a service module (e.g., *LinnAirable*) simply ensure that the `CONTROL_POINT_NAME` parameter of both modules is identical. The driver will handle the rest!

In this section, instructions will be given on how to setup these service modules. There is flexibility in how these are implemented in a CCP’s GUI, but a general structure will have to be adhered to for each module so as to ensure a smooth user experience. The demo GUI project is a great reference for how to implement these add-on modules.

## Airable Radio – *LinnAirable* Module

Airable radio is a popular internet streaming platform that grants customers live access to radio stations around the world. It's inclusion in this driver allows for browsing of several categories and a search feature to quickly find a desired radio station. Once found, a station can be selected for playback, favourited/unfavourited, or added to one of the active room's pins. There is a set structure for how this service will be set up in a project that should be followed.

On startup, the service module will register itself with the driver and then find the control point that it will be associated with by looking for the matching control point name defined in the module parameters. Some key signals regarding the status of the service module are given at the top of the module as shown in Fig. 8.

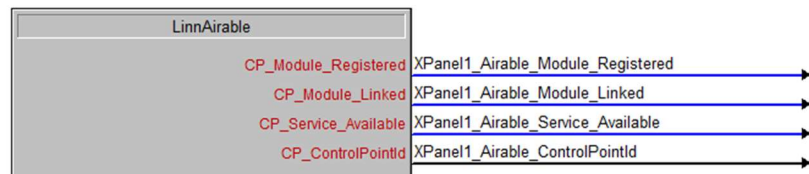


Fig. 8 – Screenshot of the first portion of signals in the *LinnAirable* SIMPL module

### Outputs

Name	Type	Description
CP_Module_Registered	Digital	On loading the program, the service module will register itself. If not high, then an error has occurred.
CP_Module_Linked	Digital	After registering, the service module will search for a control point to associate itself with. If this step fails, this signal will remain low and indicates an error has occurred. Ensure that the CONTROL_POINT_NAME parameters are identical and unique from other CCPs.
CP_Service_Available	Digital	Indicates if the airable service is available for the actively selected room on the CCP. Digital high if so, low if not. <b>Should always be available as long as your Linn DS device is registered with a Linn account.</b>
CP_ControlPointId	String	The name of the control point as set through the CONTROL_POINT_NAME parameter.

## Browse Page

The browse page of the Airable service module is shown in Fig. 9. It can be seen that there is a list of categories, or *containers*, in a list down the left-hand side. These contain collections of radio stations made by Airable. The favourites container has the stations favoured by the user on the Airable account associated with the Linn DS. The favourites container is selected automatically on the selection of a room and will populate the item list on the right. Selecting a different container from the list will select it as the new *active container*. This will load a new page of items on the right-hand side. Up to 20 items are viewable in the page. If the container has more than 20 items in it, the page previous/next buttons will become available, and you will be able to move forward and backwards through pages to view more items.

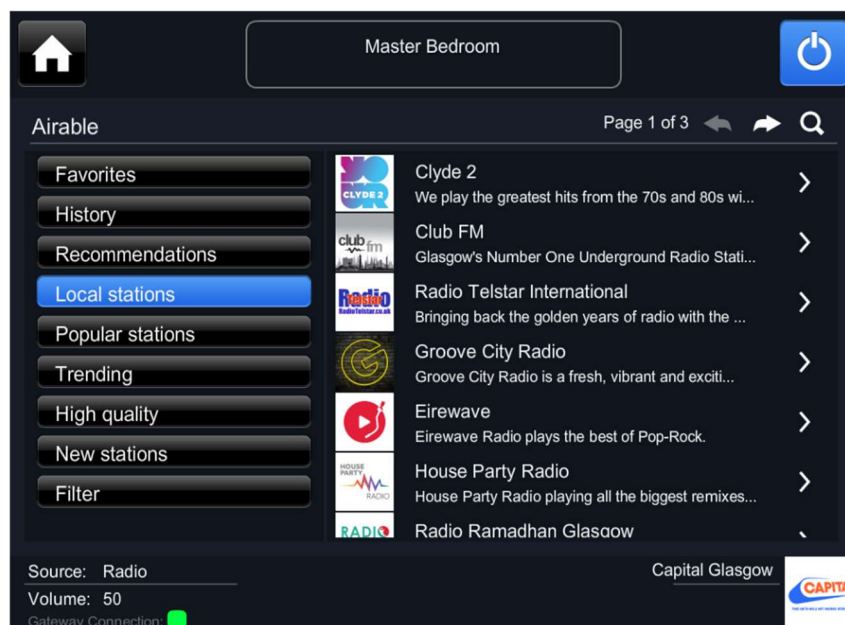


Fig. 9 – Screenshot of the browse page for Airable in the demo project. Down the left-hand side of the page are the containers – i.e., some useful categories for browsing and exploring radio stations. The lists of items on the right side will populate with stations from the active container. A maximum of 20 items is viewable. A pagination system is used when more than 20 items are available, and the user can move back and forward through pages using the buttons at the top. The right arrow visible next to each item opens up the “action page” which will be explained shortly.

## Inputs

Name	Type	Description
CP_Browse	Digital	Perform a page flip to the Airable browse page.
CP_Browse_Container[1-20]_Select	Digital	Select a container for browsing its contents.
CP_Browse_Return	Digital	Used in the ‘filter’ container. Returns to the previous level as you move through the filter navigation.
CP_Browse_PrevPage	Digital	Change to the previous page in the currently selected container.
CP_Browse_NextPage	Digital	Change to the next page in the currently selected container.
CP_Browse_List[1-20]_Select	Digital	Select an item within the active container for playback.
CP_Browse_List[1-20]_Action	Digital	Select an item within the active container as the active action item and performs a page flip to the Airable action page.

## Outputs

Name	Type	Description
CP_Browse_PageFlip	Digital	Will momentarily go high on pressing the <i>CP_Browse</i> input to signal a page flip.
CP_Browse_NumberOfContainers	Analog	The number of containers found for the Airable service.
CP_Browse_Container[1-20]_Selected	Digital	Whichever container is currently selected will be shown in this set of digital outputs. Gives feedback for the container list by allowing it to highlight the active container.
CP_Browse_Container[1-20]_Name	String	Names of the available containers for the Airable service.
CP_Browse_Return_Available	Digital	Availability of the return button.
CP_Browse_Header	String	Used in the 'filter' container. Contains info about the navigation through this container.
CP_Browse_PrevPage_Available	Digital	Availability of the previous page button. For example, when on the first page the previous page button will be unavailable.
CP_Browse_NextPage_Available	Digital	Availability of the next page button. For example, when on the last page the next page button will be unavailable.
CP_Browse_PageInfo	String	Information about the current page for the active container. E.g., "Page 1 of 5".
CP_Browse_CurrentPage	Analog	The current page number for the active container.
CP_Browse_MaxPage	Analog	The maximum page number for the active container.
CP_Browse_PageSize	Analog	The number of items in the current page for the active Airable container.
CP_Browse_List[1-20]_Line1	String	The names of the items for the current page of the active Airable container.
CP_Browse_List[1-20]_Line2	String	The descriptions of the items for the current page of the active Airable container.
CP_Browse_List[1-20]_ArtworkUri	String	The artwork URIs of the items for the current page of the active Airable container.
CP_Browse_List[1-20]_Action_Visible	Digital	Visibility of the action buttons. Used to hide the action buttons for the <i>By Location/Language/Genre</i> filter items.

## Search Page

For finding a specific station from Airable’s library, a search function is available. In the demo project a page is dedicated to this functionality that is brought up by pressing the magnifying glass button seen in the top right of Fig. 9. From the search page, shown in Fig. 10, the user can enter a search argument in the text entry box and, after hitting enter, the list below will populate with results. There is pagination available for search results and a maximum of 20 results is displayed per page. The top 3 items in the search results are filters that can also be used to refine the search. Here again, each item can be selected directly for playback, or the right arrow button can be pressed to load the item into the *action page*.

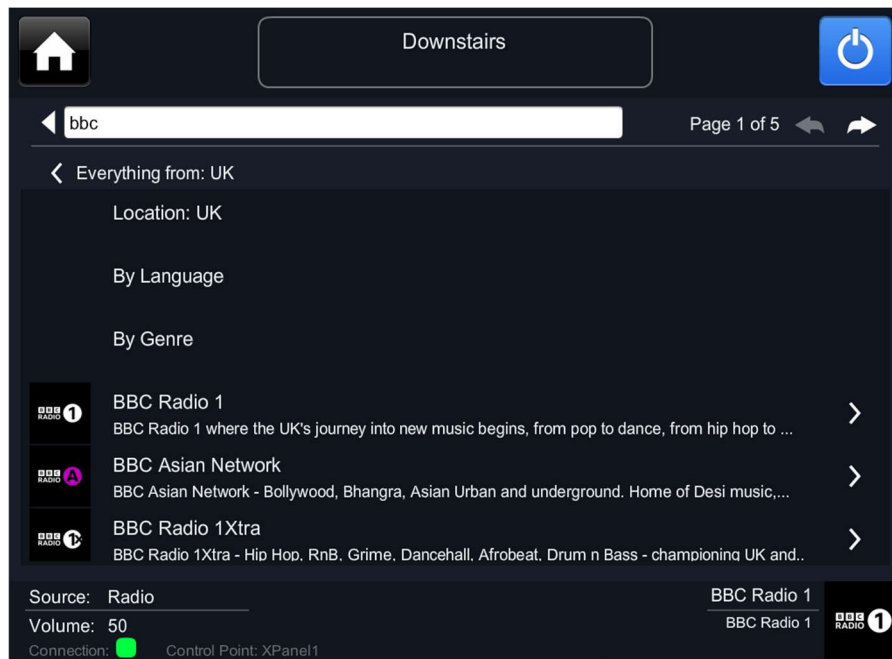


Fig. 10 – Screenshot of the search page for Airable in the demo project. The search entry bar at the top is used to enter a search argument and, after pressing enter, 20 search results will load. The *By Location/Language/Genre* items can be used to apply filters on the results – here a filter for stations from the UK has been applied. The right arrow on the result items is used to load the item the action page.

## Inputs

Name	Type	Description
CP_Search	Digital	Perform a page flip to the Airable search page.
CP_Search_TextEntry	String	Enter text for the search of the Airable service. Search is not begun until the digital signal below is made high.
CP_Search_New	Digital	Begin the search process using the string value entered in the string signal above.
CP_Search_Return	Digital	Used in the ‘filter’ functionality. Returns to the previous level as you move through the filter navigation.
CP_Search_PrevPage	Digital	Change to the previous page in the active search.
CP_Search_NextPage	Digital	Change to the next page in the active search.
CP_Search_List[1-20]_Select	Digital	Select an item from the search results for further browsing or playback.
CP_Search_List[1-20]_Action	Digital	Select an item from the search results for use by the action page.



## Outputs

<b>Name</b>	<b>Type</b>	<b>Description</b>
CP_Search_PageFlip	Digital	Will momentarily go high on pressing the “CP_Search” input to signal a page flip.
CP_Search_Return_Available	Digital	Defines the availability of the return button.
CP_Search_Header	String	Used in the ‘filter’ container. Contains info about the navigation through this container.
CP_Search_PrevPage_Available	Digital	Defines the availability of the previous page button. For example, when on the first page the previous page button will be unavailable.
CP_Search_NextPage_Available	Digital	Defines the availability of the next page button. For example, when on the last page the next page button will be unavailable.
CP_Search_PageInfo	String	Information about the current page for the active search. E.g., “Page 1 of 5”.
CP_Search_CurrentPage	Analog	The current page number for the active container.
CP_Search_MaxPage	Analog	The maximum page number for the active container.
CP_Search_PageSize	Analog	The number of items in the active search result.
CP_Search_List[1-20]_Line1	String	The names of the items in the active search result.
CP_Search_List[1-20]_Line2	String	The descriptions of the items in the active search result.
CP_Search_List[1-20]_ArtworkUri	String	The artwork URIs of the items in the active search result.
CP_Search_List[1-20]_Action_Visible	Digital	Visibility of the action buttons. Used to hide the action buttons for the <i>By Location/Language/Genre</i> filter items.

## Actions Page

A dedicated button is given to directly select stations for playback in the main page and search page. Additional options for what to do with the station are made possible through the action page. When the action input signal is used for an item, it will be loaded to the action page as the active item and a page flip will be signalled to go to the Airable action page. This page, shown in Fig. 11, gives option to play now, toggle the favourite status or set the station to a pin. The item currently selected to perform an action on is displayed at the top of the page.

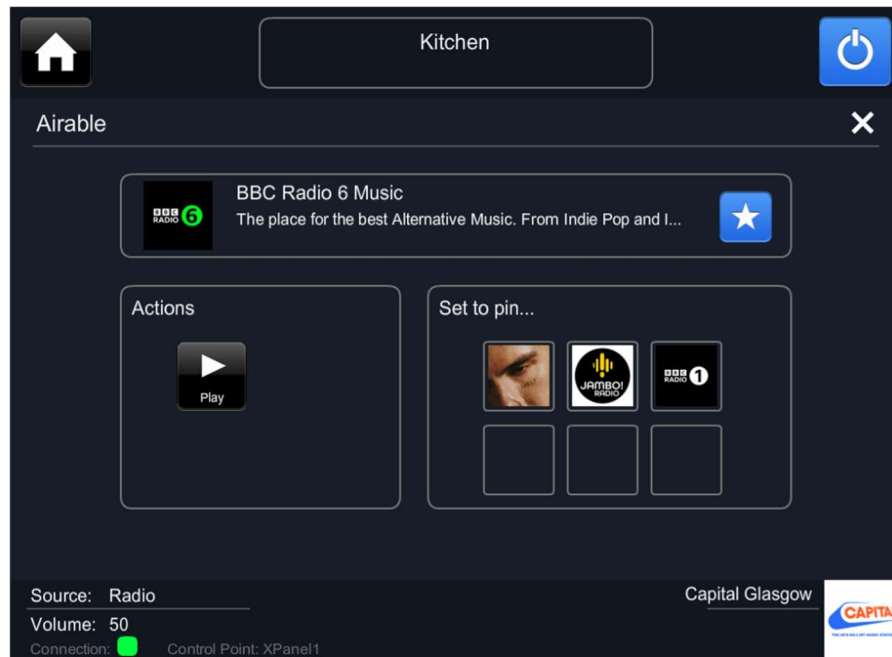


Fig. 11 – Screenshot of the action page for Airable in the demo project. The name, description and artwork for the active action item are shown at the top. Dedicated buttons to play now, and favourite/unfavourite are seen below. The item can also be set any of the pins for the active room. The artwork URI signals from the control point module are used to show the items currently occupying the six pins. The 'X' button returns the user to the previous page whether that be the browse or search page.

## Inputs

Name	Type	Description
CP_Actions_Return	Digital	Perform a page flip back to either the browse or search pages depending on the source of the active action item.
CP_Actions_PlayNow	Digital	Select the active action item for playback on the active room.
CP_Actions_Favourite_Toggle	Digital	Toggle the favourite status of the active action item.
CP_Actions_SetToPin[1-6]	Digital	Set the active action item to a pin on the active room.

## Outputs

Name	Type	Description
CP_Actions_PageFlip	Digital	Will momentarily go high on pressing the actions button for an Airable item to signal a page flip.
CP_Actions_Favourite_Status	Digital	Favourite status of the active action item.
CP_Actions_Line1	String	The name of the active action item.
CP_Actions_Line2	String	The description of the active action item.
CP_Actions_ArtworkUri	String	The artwork URI of the active action item.

The only parameter for the Airable module is the CONTROL\_POINT\_NAME:

Name	Description
CONTROL_POINT_NAME	A name that matches the name of the associated control point module.

## TIDAL – *LinnTidal* Module

TIDAL is a popular subscription-based music streaming service that allows customers to browse and play music from an extensive catalogue. This service can be incorporated into a project using the *LinnTidal* SIMPL module and customers are capable of exploring a number of categories created by TIDAL or searching for their favourite music directly. Once found, an album/playlist/track can be added to the DS playlist for playback, favourited/unfavourited, or added to one of the active room's pins. There is a set structure for how this service should be set up in a project that should be followed for a smooth user experience.

On startup, the service module will register itself with the driver and then find the control point that it will be associated with by looking for the matching control point name defined in the module parameters. To login to the Qobuz service, load up the gateway API UI by loading the page [http://\[location\]:4100/](http://[location]:4100/) where [location] is the IP address of the DS running the gateway or the device running Kazoo Server. Some key signals regarding the status of the service module are give at the top of the module as shown in Fig 12.

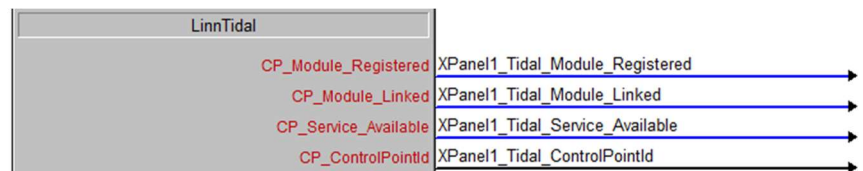


Fig. 12 – Screenshot of the first portion of signals in the *LinnTidal* SIMPL module

### Outputs

Name	Type	Description
CP_Module_Registered	Digital	On loading the program, the service module will register itself. If not high, then an error has occurred.
CP_Module_Linked	Digital	After registering, the service module will search for a control point to associate itself with. If this step fails, this signal will remain low and indicates an error has occurred. Ensure that the CONTROL_POINT_NAME parameters are identical and unique from other CCPs.
CP_Service_Available	Digital	Used to indicate if the TIDAL service is available for the actively selected room on the CCP. Digital high if so, low if not
CP_ControlPointId	String	The name of the control point as set through the CONTROL_POINT_NAME parameter.

## Home Page

The first point of entry for TIDAL functionality should be a “home page” where a number of tabs are given as options for browsing. A shortcut to reach the search page is also shown in this page. The TIDAL home page for the demo project is shown in Fig 13.

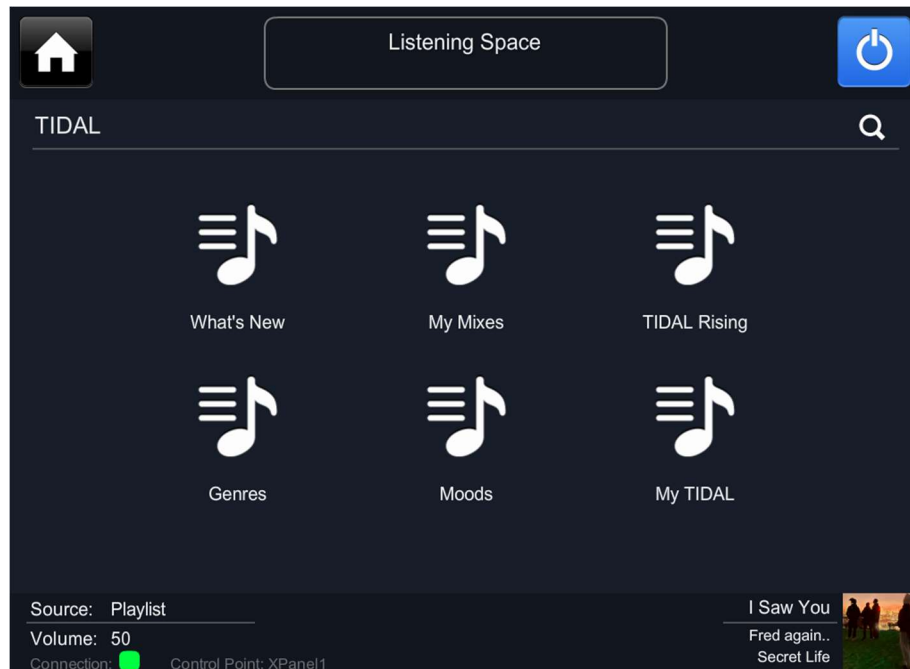


Fig. 13 – Screenshot of the home page for TIDAL in the demo project. The 6 top-level tabs of the service are shown and a shortcut to the search page is in the top-right.

## Inputs

Name	Type	Description
CP_Home	Digital	Perform a page flip to the TIDAL home page.
CP_WhatsNew	Digital	Select the “What’s New” TIDAL tab as the active tab and perform a page flip to the browse page.
CP_MyMixes	Digital	Select the “My Mixes” TIDAL tab as the active tab and performs a page flip to the browse page.
CP_TidalRising	Digital	Selects the “TIDAL Rising” TIDAL tab as the active tab and perform a page flip to the browse page.
CP_Genres	Digital	Select the “Genres” TIDAL tab as the active tab and perform a page flip to the browse page.
CP_Moods	Digital	Select the “Moods” TIDAL tab as the active tab and perform a page flip to the browse page.
CP_MyTidal	Digital	Select the “My TIDAL” TIDAL tab as the active tab and perform a page flip to the browse page.
CP_Search	Digital	Perform a page flip to the TIDAL search page.

## Outputs

Name	Type	Description
CP_Home_PageFlip	Digital	Will momentarily go high on pressing the <i>CP_Home</i> input to signal a page flip.
CP_WhatsNew_Available	Digital	Availability of the “What’s New” TIDAL tab for the active room.
CP_MyMixes_Available	Digital	Availability of the “My Mixes” TIDAL tab for the active room.

CP_TidalRising_Available	Digital	Availability of the “TIDAL Rising” TIDAL tab is available for the actively selected room on the CCP.
CP_Genres_Available	Digital	Availability of the “Genres” TIDAL tab for the active room.
CP_Moods_Available	Digital	Availability of the “Moods” TIDAL tab for the active room.
CP_MyTidal_Available	Digital	Availability of the “My TIDAL” TIDAL tab for the active room.

## Browse Page

On selecting a TIDAL tab, the demo project will perform a page flip to the browse page where the tab will be loaded and ready for exploring further. An example of what this looks like for the “What’s New” tab is shown in Fig. 14 and Fig. 15.

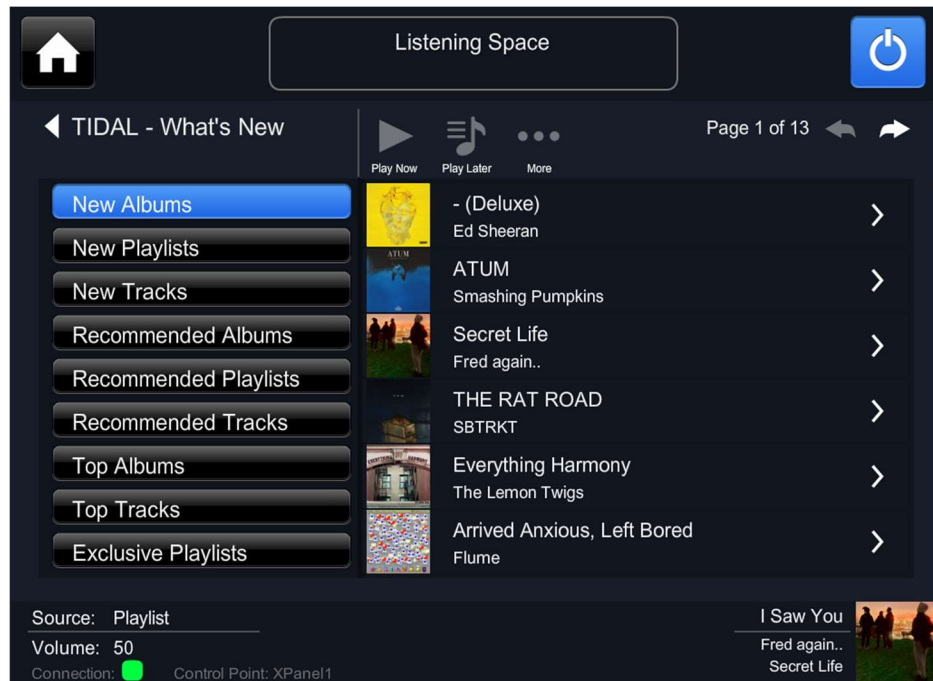


Fig. 14 – Screenshot of the browse page for TIDAL in the demo project. The “New Albums” container is the active container, and a list of collections is being displayed on the right. The action buttons (play now, play later, more) are unavailable for collections. The usual pagination buttons and info are displayed in the top-right and the right arrows will load the collection to the actions page.

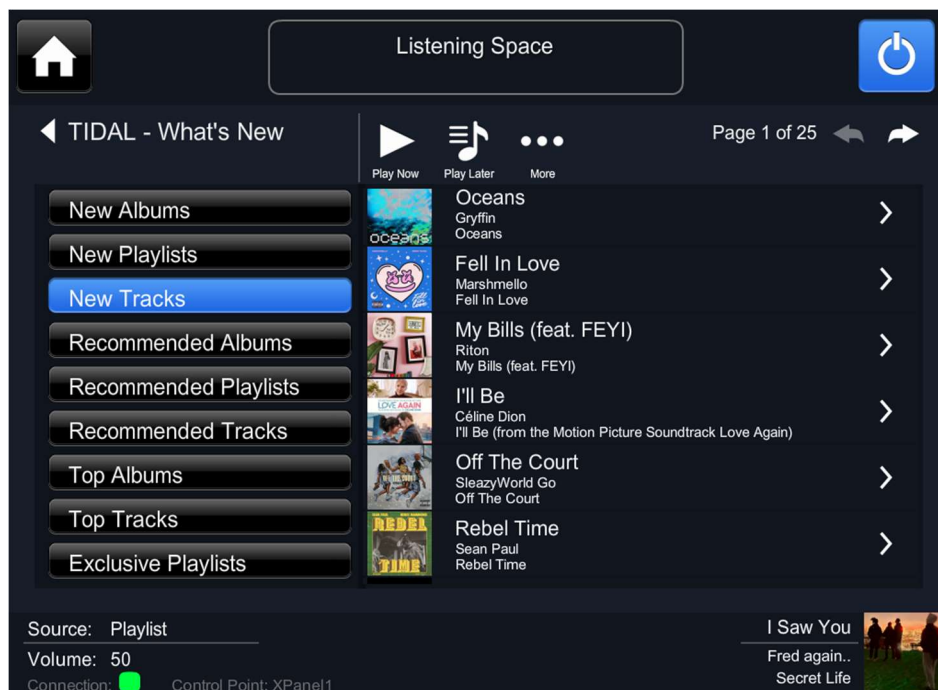


Fig. 15 – Screenshot of the browse page for TIDAL in the demo project. The “New Tracks” container is the active container, and a list of items is being displayed on the right. The action buttons (play now, play later, more) are available for items and will load the entire contents of the container for playback. The usual pagination buttons and info are displayed in the top-right and the right arrows will load the item to the actions page.

Similar in design to the browse page of the airable service, this page allows for browsing of the active tab. The list on the left side gives a selection of *containers* to select within the tab. On selecting a container, it will be set as the *active container* and its contents will be displayed in the list on the right. The contents of the list are flexible to either show *items* (tracks) or *collections* (albums and playlists). In the above example the New Albums container shows a list of albums and hence the collection signals are visible. If we were to select the “New Tracks” container, then the right list will display items instead. This results in an extra line of information being displayed as shown in Fig 15. Selecting an item will select it for playback whereas selecting a collection will load its contents in the collection page.

#### Inputs

Name	Type	Description
CP_Browse_ContainerReturn	Digital	Return to the main container list for the active tab. Used in the occasional case where a tab has sub-containers (in the TIDAL “Genres” tab for instance).
CP_Browse_Container[1-20]_Select	Digital	Select a container for browsing its contents.
CP_Browse_Active_ClearAndPlay	Digital	Clears the DS playlist and loads the contents of the active container for playback.
CP_Browse_Active_PlayLater	Digital	Load the contents of an active container for playback at the end of the DS playlist.
CP_Browse_Active_Action	Digital	Load the contents of an active container to the actions page and performs a page flip to the TIDAL action page.
CP_Browse_PrevPage	Digital	Change to the previous page in the active container.
CP_Browse_NextPage	Digital	Change to the next page in the active container.
CP_Browse_List[1-20]_Item_Select	Digital	Select an item within the active container for playback.
CP_Browse_List[1-20]_Collection_Select	Digital	Load the contents of a collection within the active container to the collection page and performs a page flip to the TIDAL collection page.
CP_Browse_List[1-20]_Action	Digital	Select a list item/collection as the active action item and performs a page flip to the TIDAL action page.

#### Outputs

Name	Type	Description
CP_Browse_PageFlip	Digital	Will momentarily go high on pressing one of the tab inputs to signal a page flip.
CP_Browse_Active_Tab_Name	String	The name of the tab currently active in the browse page.
CP_Browse_ContainerReturn_Available	Digital	Availability of the container return button. Becomes unavailable when it is not relevant.
CP_Browse_Active_Container_Name	String	The name of the container currently active in the browse page.
CP_Browse_NumberOfContainers	Analog	The number of containers found for the active tab of the TIDAL service.



CP_Browse_Container[1-20]_Selected	Digital	The currently selected container will be shown in this set of digital outputs. Gives feedback for the container list by allowing it to highlight the active container.
CP_Browse_Container[1-20]_Name	String	Names of the available containers for the active tab of the service.
CP_Browse_Active_Actions_Available	Digital	Availability of the action buttons. Makes them unavailable when a container of collections is active.
CP_Browse_PrevPage_Available	Digital	Availability of the previous page button. For example, when on the first page the previous page button will be unavailable.
CP_Browse_NextPage_Available	Digital	Availability of the next page button. For example, when on the last page the next page button will be unavailable.
CP_Browse_PageInfo	String	Information about the current page for the active container. E.g., "Page 1 of 5".
CP_Browse_CurrentPage	Analog	The current page number for the active container.
CP_Browse_MaxPage	Analog	The maximum page number for the active container.
CP_Browse_PageSize	Analog	The number of list items/collections in the current page for the active container.
CP_Browse_List_Items_Visible	Digital	Visibility of the list items. Will hide the buttons and labels when the active container contains collections.
CP_Browse_List[1-20]_Item_Line1	String	The track names of the items for the current page of the active container.
CP_Browse_List[1-20]_Item_Line2	String	The artist names of the items for the current page of the active container.
CP_Browse_List[1-20]_Item_Line3	String	The album names of the items for the current page of the active container.
CP_Browse_List_Collections_Visible	Digital	Visibility of the list collections. Will hide the buttons and labels when the active container contains items.
CP_Browse_List[1-20]_Collection_Line1	String	The album/playlist names of the collections for the current page of the active container.
CP_Browse_List[1-20]_Collection_Line2	String	The album artists or playlist descriptions of the collections for the current page of the active container.
CP_Browse_List[1-20]_ArtworkUri	String	The artwork URIs of the items/collections for the current page of the active container.

## Search Page

From the TIDAL home page, you can also access a search page which allows customers to search the streaming service's full catalogue. The search page can be accessed in the demo project from the home page via the magnifying glass in the top-right of Fig. 13 and a screenshot of this page is shown in Fig. 16. The user can enter a search argument in the text entry box and, after hitting enter, the list below will populate with results. You can select whether you want artists, tracks, albums, or playlist results and there is pagination to view more than the default 20 results if necessary. Artists is selected initially on a new search. Here again each item or collection can be selected for playback or further browsing. The right arrow action button can also be pressed to load the item/collection into the action page.

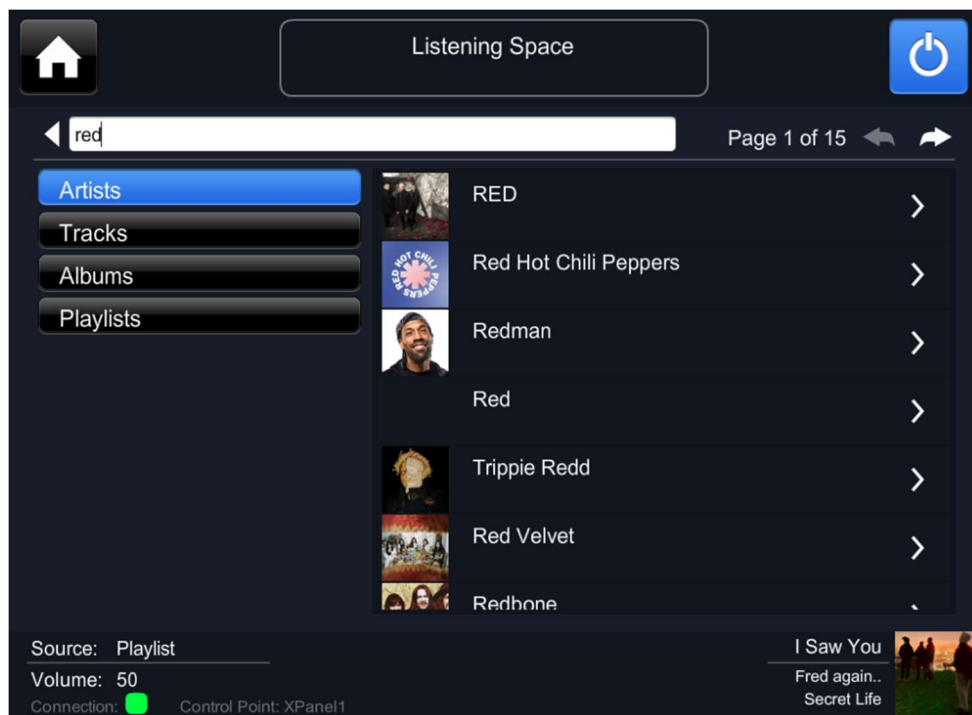


Fig. 16 – Screenshot of the search page for TIDAL in the demo project. A search has been entered in the text entry bar and you can choose whether you want artists, tracks, albums, or playlists for the results in the right list. The usual pagination buttons and info are displayed in the top-right and the right arrows will load the items/collections to the actions page.

## Inputs

Name	Type	Description
CP_Search_TextEntry	String	Enter text for the search of the TIDAL service. Search is not begun until the digital signal below is made high.
CP_Search_New	Digital	Begin the search process using the string value entered in the string signal above.
CP_Search_Artists	Digital	Select artists as the active type of search results.
CP_Search_Tracks	Digital	Select tracks as the active type of search results.
CP_Search_Albums	Digital	Select albums as the active type of search results.
CP_Search_Playlists	Digital	Select playlists as the active type of search results.

CP_Search_PrevPage	Digital	Change to the previous page in the active search.
CP_Search_NextPage	Digital	Change to the next page in the active search.
CP_Search_List[1-20]_Item_Select	Digital	Select an item within the active search for playback.
CP_Search_List[1-20]_Collection_Select	Digital	Load the contents of a collection within the active search to the collection page and performs a page flip to the TIDAL collection page.
CP_Search_List[1-20]_Action	Digital	Select a list item/collection as the active action item and performs a page flip to the TIDAL action page.

## Outputs

Name	Type	Description
CP_Search_PageFlip	Digital	Will momentarily go high when a page flip to the Search page is required.
CP_Search_Artists_Selected	Digital	Feedback that the artists result type is the active search type.
CP_Search_Tracks_Selected	Digital	Feedback that the tracks result type is the active search type.
CP_Search_Albums_Selected	Digital	Feedback that the albums result type is the active search type.
CP_Search_Playlists_Selected	Digital	Feedback that the playlists result type is the active search type.
CP_Search_Types_Available	Digital	Availability of the search result type buttons. Unavailable until a search is started.
CP_Search_PrevPage_Available	Digital	Availability of the previous page button. For example, when on the first page the previous page button will be unavailable.
CP_Search_NextPage_Available	Digital	Availability of the next page button. For example, when on the last page the next page button will be unavailable.
CP_Search_PageInfo	String	Information about the current page for the active search. Eg. "Page 1 of 5".
CP_Search_CurrentPage	Analog	The current page number for the active search. Will display zero if no search is active.
CP_Search_MaxPage	Analog	The maximum page number for the active search. Will display zero if no search is active.
CP_Search_PageSize	Analog	The number of items in the current page for the active search.
CP_Search_List_Items_Visible	Digital	Visibility of the list items. Use to hide the items info/buttons when the active search contains collections.
CP_Search_List[1-20]_Item_Line1	String	The track names of the items for the current page of the active search.

CP_Search_List[1-20]_Item_Line2	String	The artist names of the items for the current page of the active search.
CP_Search_List[1-20]_Item_Line3	String	The album names of the items for the current page of the active search.
CP_Search_List[1-20]_Collections_Visible	Digital	Visibility of the list collections. Use to hide the collections info/buttons when the active search contains items.
CP_Search_List[1-20]_Collection_Line1	String	The album/playlist/artist names of the collections for the current page of the active search.
CP_Search_List[1-20]_Collection_Line2	String	The album artists or playlist descriptions of the collections for the current page of the active search.
CP_Search_List[1-20]_ArtworkUri	String	The artwork URIs of the items/collections for the current page of the active search.

## Collection Page

When a collection (a playlist or album) is selected from the browse or search pages the collection is loaded to the collection page and a page flip is prompted. This allows you to view the contents of the collection and perform actions with it. An example of the collection page is shown in Fig. 17

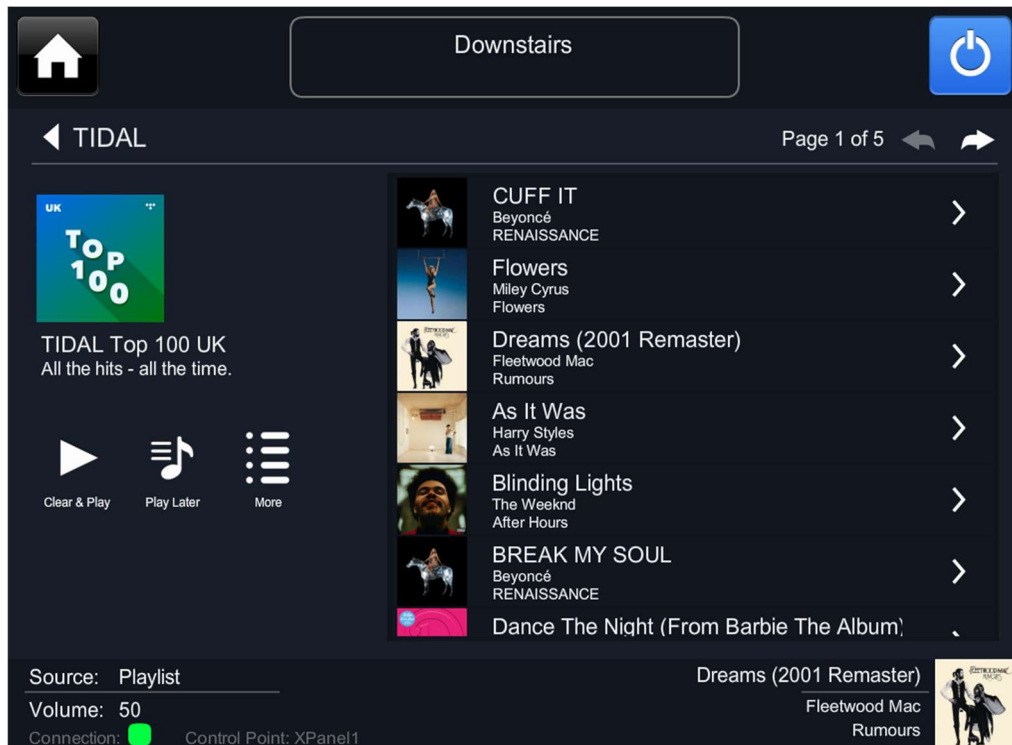


Fig. 17 – Screenshot of the collection page for TIDAL in the demo project. The collection is the “TIDAL Top 100 UK” playlist and the whole collection can be selected/queued for playback through the “Clear & Play” or “Play Later” buttons or loaded to the action page by pressing the “More” button. The contents of the collection are viewed in the right list and can be individually selected for playback. The usual pagination buttons and info are displayed in the top-right and the right arrows will load the individual items to the actions page.

## Inputs

Name	Type	Description
CP_Collection_Return	String	Perform a page flip back to either the browse or search pages depending on the source of the active collection.
CP_Collection_Active_ClearAndPlay	Digital	Clears the DS playlist and loads the contents of the active collection for playback.
CP_Collection_Active_PlayLater	Digital	Load the contents of an active collection for playback at the end of the DS playlist.
CP_Collection_Active_Action	Digital	Load the active collection to the actions page and performs a page flip to the TIDAL action page.
CP_Collection_PrevPage	Digital	Change to previous page in the active collection.
CP_Collection_NextPage	Digital	Change to next page in the active collection.
CP_Collection_Item[1-20]_Select	Digital	Select an item from the collection for playback.
CP_Collection_Item[1-20]_Action	Digital	Load an item from the collection to the actions page.

## Outputs

Name	Type	Description
CP_Collection_PageFlip	Digital	Will momentarily go high when a page flip to the Collection page is required.
CP_Collection_Active_Actions_Available	Digital	Availability of the action buttons.
CP_Collection_PrevPage_Available	Digital	Availability of the previous page button. For example, when on the first page the previous page button will be unavailable.
CP_Collection_NextPage_Available	Digital	Availability of the next page button. For example, when on the last page the next page button will be unavailable.
CP_Collection_PageInfo	String	Information about the current page for the active collection. Eg. "Page 1 of 5".
CP_Collection_CurrentPage	Analog	The current page number for the active collection. Will display zero if no collection is active.
CP_Collection_MaxPage	Analog	The maximum page number for the active collection. Will display zero if no collection is loaded.
CP_Collection_PageSize	Analog	The number of items in the current page for the active collection.
CP_Collection_Line1	String	The album/playlist name of the active collection.
CP_Collection_Line2	String	The album artist or playlist description of the active collection.
CP_Collection_ArtworkUri	String	The artwork URI of the active collection.
CP_Collection_Item[1-20]_Line1	String	The track names of the items for the current page of the active collection.
CP_Collection_Item[1-20]_Line2	String	The artist names of the items for the current page of the active collection.
CP_Collection_Item[1-20]_Line3	String	The album names of the items for the current page of the active collection.
CP_Collection_Item[1-20]_ArtworkUri	String	The artwork URIs of the items/collections for the current page of the active collection.

## Actions Page

When the action button is pressed for item or collection, it will be loaded as the active action item and page flip will occur to the action page as shown in Fig. 18. Further options for playback are given here as well as option to favourite/unfavourite the item/collection and set it to a pin.

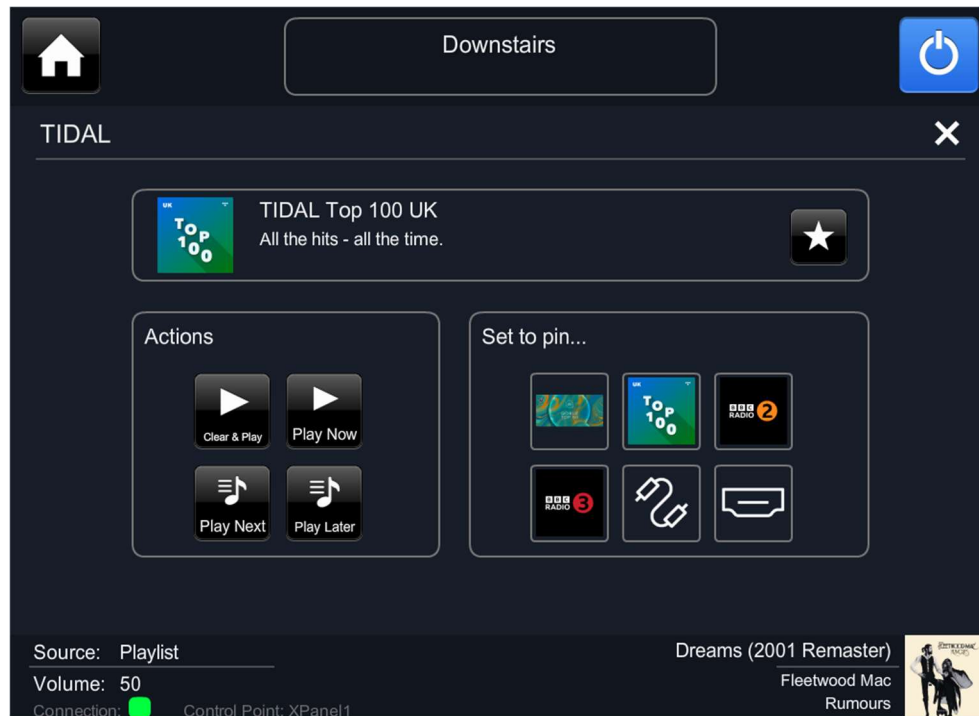


Fig. 18 – Screenshot of the actions page for TIDAL in the demo project. The active action item is the “TIDAL Top 100 UK” playlist and the whole collection can be selected/queued for playback through the “Play Now” or “Play Later” buttons or loaded to the action page. The contents of the collection are viewed in the right list and can be individually selected for playback. The usual pagination buttons and info are displayed in the top-right and the right arrows will load the items to the actions page.

## Inputs

Name	Type	Description
CP_Actions_Return	Digital	Perform a page flip back to either the browse, search or collection pages depending on the source of the active action item.
CP_Actions_ClearAndPlay	Digital	Clears the DS playlist and loads the active active action item for playback.
CP_Actions_PlayNow	Digital	Load the active action item to the DS playlist and select for playback
CP_Actions_PlayNext	Digital	Load the active action item after the currently playing track in the DS playlist.
CP_Actions_PlayLater	Digital	Load the active action item at the end of the DS playlist.
CP_Actions_Favourite_Toggle	Digital	Toggles the favourite status of the active action item.
CP_Actions_SetToPin[1-6]	Digital	Set the active action item to a pin on the active room.

## Outputs

<b>Name</b>	<b>Type</b>	<b>Description</b>
CP_Actions_PageFlip	Digital	Will momentarily go high on pressing the actions button for a TIDAL item/collection to signal a page flip.
CP_Actions_Favourite_Status	Digital	Favourite status of the active action item.
CP_Actions_Favourite_Available	Digital	Availability of the favourite button for the active action item. Some TIDAL items such as artists cannot be favourited.
CP_Actions_Line1	String	The name of the active action item.
CP_Actions_Line2	String	The artist's name or playlist description of the active action item.
CP_Actions_Line3	String	The album name of the active action item.
CP_Actions_ArtworkUri	String	The artwork URI of the active action item.



## Qobuz – *LinnQobuz* Module

Qobuz is a popular subscription-based music streaming service that allows customer to browse and play music from an extensive catalogue. This service can be incorporated into a project using the *LinnQobuz* SIMPL module and customers are able to explore a number of categories created by Qobuz or search for their favourite music directly. Once found, an album/playlist/track can be added to the DS playlist for playback, favourited/unfavourited, or added to one of the active room's pins. There is a set structure for how this service should be set up in a project that should be followed for a smooth user experience. **This structure is very similar to that used for TIDAL support so the previous section will be referred to so as to avoid repeating instructions.**

On startup the driver will register the service module and find the control point that it will be associated with by looking for the matching control point name in the module parameters. To login to the Qobuz service load up the gateway API UI by loading the page [http://\[location\]:4100/](http://[location]:4100/) where [location] is the IP address of the DS running the gateway or the device running Kazoo Server. Some key signals regarding the status of the service module are given at the top of the module as shown in Fig 19.

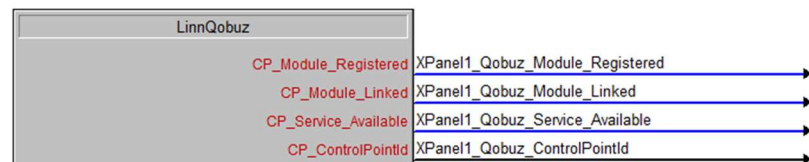


Fig. 19 – Screenshot of the first portion of signals in the *LinnQobuz* SIMPL module

### Outputs

Name	Type	Description
CP_Module_Registered	Digital	On loading the program, the service module will register itself. If not high, then an error has occurred.
CP_Module_Linked	Digital	After registering, the service module will search for a control point to associate itself with. If this step fails, this signal will remain low and indicates an error has occurred. Ensure that the CONTROL_POINT_NAME parameters are identical and unique from other CCPs.
CP_Service_Available	Digital	Used to indicate if the Qobuz service is available for the actively selected room on the CCP. Digital high if so, low if not
CP_ControlPointId	Digital	The name of the control point as set through the CONTROL_POINT_NAME parameter.

## Home Page

The first point of entry for Qobuz functionality should be a “home page” where a number of tabs are given as options for browsing. A shortcut to reach the search page is also shown in this page. The Qobuz home page for the demo project is shown in Fig. 20. This is similar to TIDAL but differs in the tabs made available by the streaming service and there are only 3 tabs rather than 6.

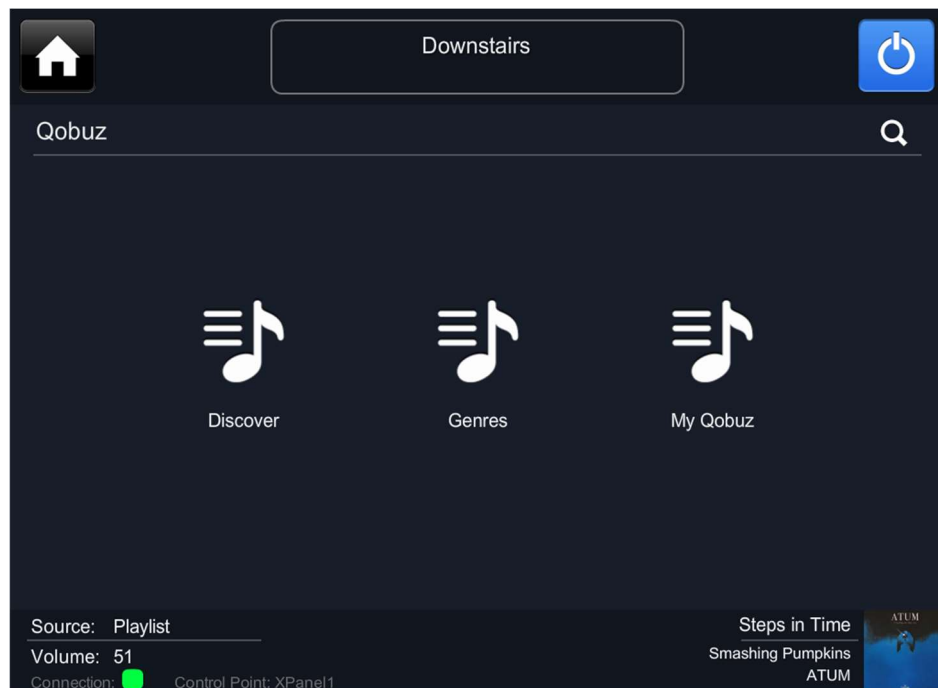


Fig. 20 – Screenshot of the home page for Qobuz in the demo project. The 3 top-level tabs of the service are shown and a shortcut to the search page is in the top-right.

## Inputs

Name	Type	Description
CP_Home	Digital	Perform a page flip to the Qobuz home page.
CP_Discover	Digital	Selects the “Discover” Qobuz tab as the active tab and performs a page flip to the browse page.
CP_Genres	Digital	Selects the “Genres” Qobuz tab as the active tab and performs a page flip to the browse page.
CP_MyQobuz	Digital	Selects the “My Qobuz” Qobuz tab as the active tab and performs a page flip to the browse page.
CP_Search	Digital	Perform a page flip to the Qobuz search page.

## Outputs

Name	Type	Description
CP_Home_PageFlip	Digital	Will momentarily go high on pressing the “CP_Home” input to perform a page flip.
CP_Discover_Available	Digital	Availability of the “Discover” Qobuz tab for the active room.
CP_Genres_Available	Digital	Availability of the “Genres” Qobuz tab for the active room.
CP_MyQobuz_Available	Digital	Availability of the “My Qobuz” Qobuz tab for the active room.

## Other Pages

The rest of the module is practically identical in structure to the TIDAL module. You have browse, search, collection, and actions pages with identical signal sets. Please see the above section on TIDAL for a guide in setting up the rest of the Qobuz streaming service. Once again, the demo project will also provide useful information for how to structure and implement this service module.

## Playlist – *LinnPlaylist* Module

Each Linn DS/DSM has its own playlist source that acts like a queue where tracks can be added, moved, removed etc from various sources (streaming services, local files servers, cloud playlists). Within the playlist you can move to a new point by selecting a different track for playback. The *LinnPlaylist* module allows for viewing and manipulation of the playlist for the active room. There is a set structure for how this service should be set up in a project and this should be followed.

On startup, the add-on module will register itself with the driver and then find the control point that it will be associated with by looking for the matching control point name defined in the module parameters. Some key signals regarding the status of the add-on module are given at the top of the module as shown in Fig. 21. Unlike the previous add-on modules there is no “service available” output as the playlist will always be available.



Fig. 21 – Screenshot of the first portion of signals in the *LinnPlaylist* SIMPL module

### Inputs

Name	Type	Description
CP_Page	Digital	Perform a page flip to the playlist page.

### Outputs

Name	Type	Description
CP_Module_Registered	Digital	On loading the program, the add-on module will register itself. If not high, then an error has occurred.
CP_Module_Linked	Digital	After registering, the add-on module will search for a control point to associate itself with. If this step fails, this signal will remain low and indicates an error has occurred. Ensure that the CONTROL_POINT_NAME parameters are identical and unique from other CCPs.
CP_ControlPointId	String	The name of the control point as set through the CONTROL_POINT_NAME parameter.
CP_PageFlip	Digital	Will momentarily go high on pressing the <i>CP_Page</i> input to signal a page flip.

## Playlist

The main page of the playlist module is shown in Fig 22. The contents of the playlist are shown in a list on the right and any changes will be promptly updated. The track that is currently selected (it could be playing or paused) is shown via a play button on top of the artwork URI for the item. The number of items shown in the playlist list is 50 by default for this module. When the playlist is larger than this the usual pagination system can be used to move between pages of items.

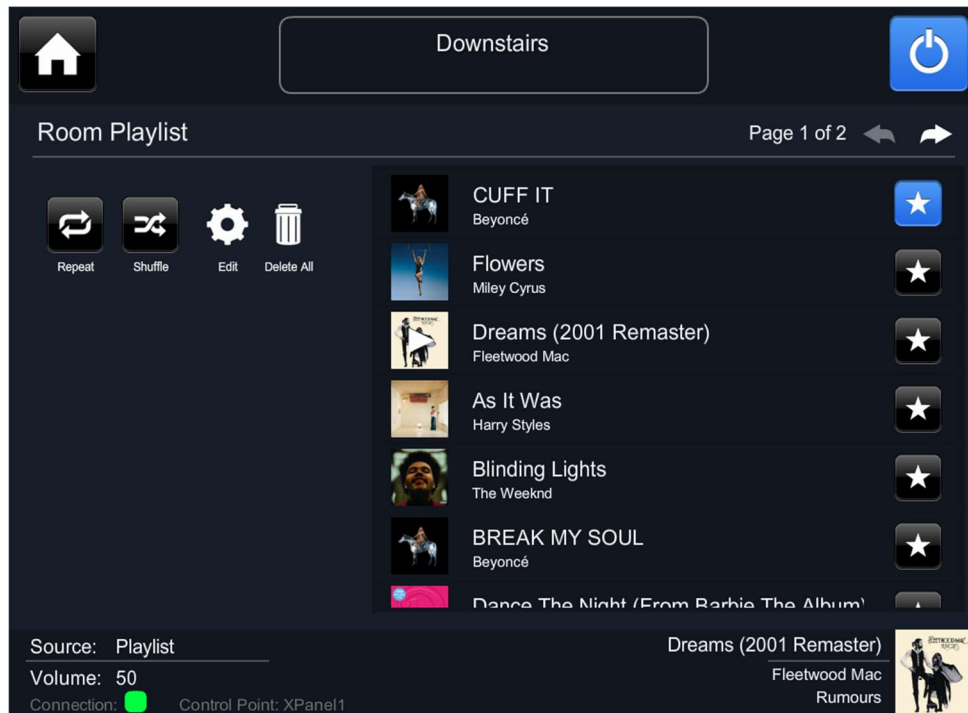


Fig. 22 – Screenshot of the page for the playlist module in the demo project. The playlist of the active room is shown in the list on the right and the favourite status of each item can be viewed and toggled. On the left are buttons to enable the *editing mode* and to clear the playlist. The usual pagination buttons and info are displayed in the top-right and can be used if the playlist contains more than 50 items. The repeat and shuffle buttons are also included in this page for the demo project.

## Inputs

Name	Type	Description
CP_PrevPage	Digital	Change to the previous page in the active room's playlist.
CP_NextPage	Digital	Change to the next page in the active room's playlist.
CP_DeleteAll	Digital	Clear all items from the active room's playlist.
CP_List[1-50]_Select	Digital	Select an item within the playlist for playback (i.e., move to a new position in the playlist). If editing is active this will be used to set the active edit item.
CP_List[1-50]_Favourite_Toggle	Digital	Toggle the favourite status of an item within the playlist.

## Outputs

Name	Type	Description
CP_PrevPage_Available	Digital	Availability of the previous page button. For example, when on the first page the previous page button will be unavailable.
CP_NextPage_Available	Digital	Availability of the next page button. For example, when on the last page the next page button will be unavailable.
CP_PageInfo	String	Information about the current page for the active room's playlist. E.g., "Page 1 of 5".
CP_CurrentPage	Analog	The current page number for the active room's playlist.
CP_MaxPage	Analog	The maximum page number for the active room's playlist.
CP_PageSize	Analog	The number of items in the current page for the active room's playlist.
CP_List[1-50]_Selected	Digital	Whichever item is currently selected in the playlist will be shown in this set of digital outputs. Gives feedback for the playlist by allowing it to highlight the track currently selected.
CP_List[1-50]_Favourite_Status	Digital	Favourite status for the playlist items within the current page.
CP_List[1-50]_Line1	String	The track name for the playlist items within the current page.
CP_List[1-50]_Line2	String	The artist's name for the playlist items within the current page.
CP_List[1-50]_ArtworkUri	String	The artwork URI for the playlist items within the current page.

## Edit

To make modifications to individual items within the playlist you must first enable the editing mode for the playlist module. This allows you to click on an item within the playlist and rather than selecting it for playback, it will highlight it and you are then able to perform a number of actions upon it. In the demo project the buttons for edit functionality are invisible until the edit mode is enabled and unavailable until an item is clicked. See Fig. 23 for an example of the edit mode enabled and an item selected for editing. Only one item can be selected for editing at a time and if the item that is already selected is pressed again it will unselect it.

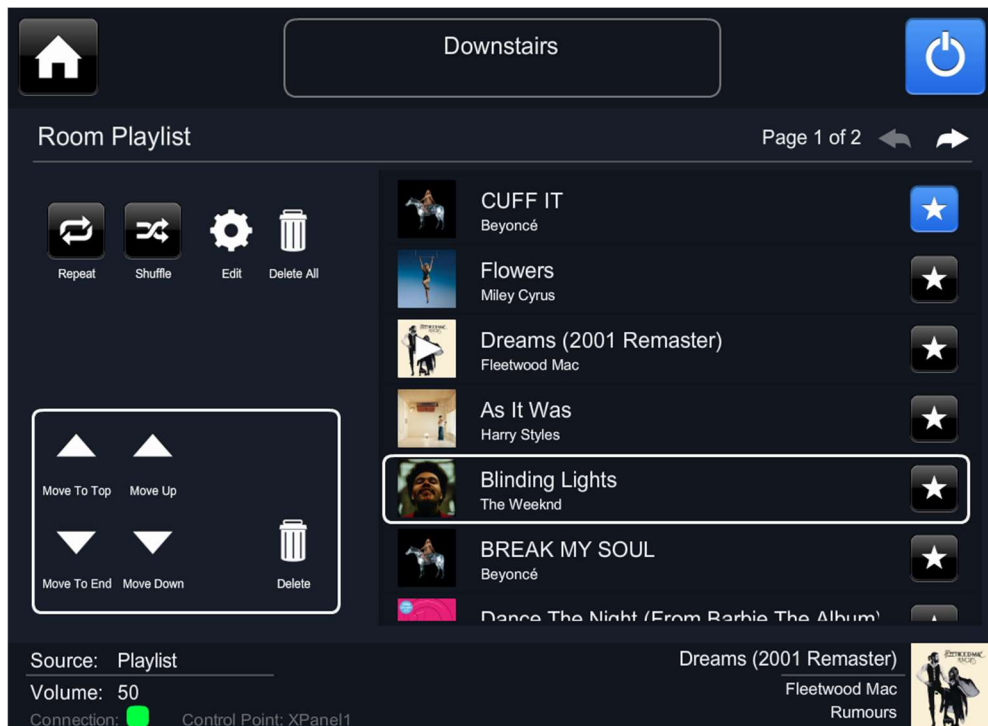


Fig. 23 – Screenshot of the page for the playlist module in the demo project with editing mode enabled. The Weeknd track within the playlist has been selected for editing and is indicated by the white outline. Further buttons are now visible in the bottom left that allow actions to be performed on the selected track. The item can be moved up or down the playlist or removed.

## Inputs

Name	Type	Description
CP_Edit_Toggle	Digital	Used to toggle the edit state of the playlist page.
CP_MoveToTop	Digital	Move the playlist item selected for editing to the top of the playlist.
CP_MoveUp	Digital	Move the playlist item selected for editing one place up the playlist.
CP_MoveDown	Digital	Move the playlist item selected for editing one place down the playlist.
CP_MoveToEnd	Digital	Move the playlist item selected for editing to the end of the playlist.
CP_Delete	Digital	Remove the playlist item selected for editing from the playlist.

## Outputs

Name	Type	Description
CP_Edit_Active	Digital	Indicates whether the edit state is active.
CP_Edit_Selected_Any	Digital	Indicates whether an item has been selected for editing in the playlist. Used for the availability of the edit inputs.
CP_Edit[1-50]_Selected	Digital	Whichever item in the active playlist list is currently selected for editing will be shown in this set of digital outputs.



## Glossary

**Active room** – A CCP can select any room from the local network as its actively selected room. This is selected through the CCP and can be changed at any time.

**Core4** – A category of Linn DS products. A list of these devices, which are capable of running the Gateway API, can be found here: <https://docs.linn.co.uk/wiki/index.php/Category:CORE4>

**Crestron control point (CCP)** – A device through which the Crestron system is controlled (eg. Touch panel, iPad, Phone, XPanel application) with a GUI designed through Crestron's software tools.

**Gateway API** – Linn's own API for managing Linn devices across a local network. A WebSockets connection is established between the Crestron control processor and the gateway API and is used to learn about the devices within the network and control features on them. Comes included with Kazoo server software and on Core4 Linn DSM products.

**Grouping** – Linn devices are capable of being grouped together to allow two or more systems to play the same music in synchronisation. One device will lead the group and all listening devices will play the same output audio.

**House Topology** – The topology of Linn devices found by the Gateway API within a local network. Contained within this will also be information about groupings of devices, i.e., if there are devices on the local network that are listening to other devices. For the Kustom DSM there will be additional information about the arrangement of streams and zones.

**Listening Room** – When Linn DS devices are grouped to play the same audio, there will be one room that leads the group. The other devices are referred to as listeners or listening rooms.

**LPEC** – Linn Protocol for Eventing and Control, an existing protocol for direct control over Linn devices: <https://docs.linn.co.uk/wiki/index.php/Developer:LPEC>

**Room** – A room in Linn DS/DSM products refers to a single playable device. The majority of products Majik/Selekt/Klimax are equivalent to a single room. This is not as simple in the case of Kustom DSM where there are multiple devices contained within the one box. For more info on how streams and zones operate see [https://docs.linn.co.uk/wiki/index.php/Kustom\\_DSM](https://docs.linn.co.uk/wiki/index.php/Kustom_DSM). More specifically, in this driver, a room refers to anything that could occupy a line on the room selection menu which could be a standalone Linn DS or a Kustom stream or zone.

**UDN** – A fixed device identifier for Linn DS devices. Can be used to ensure that rooms for device locking can be found even if their visible name is changed through system configuration. Can be found by looking through the Crestron control processor logs on start-up and looking for the */V2/topology/status* response.